FIG. 1

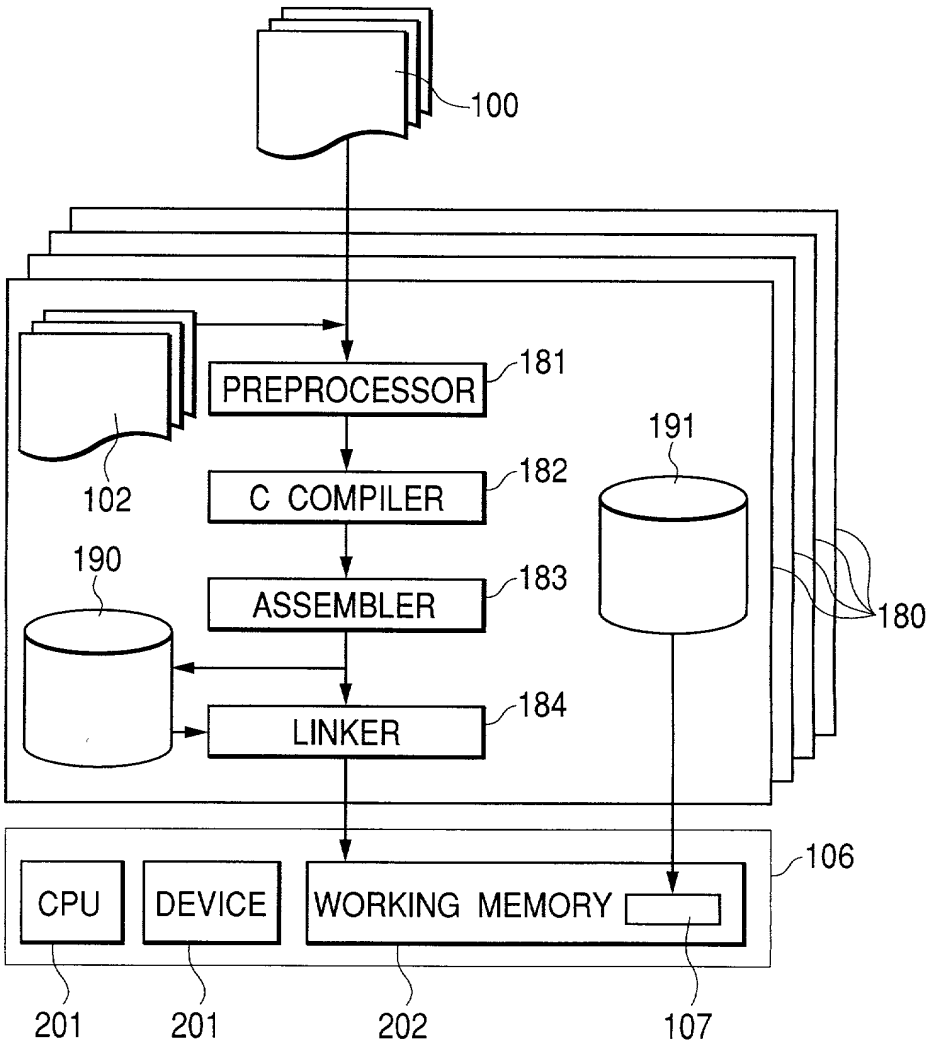
100
PREPROCESSOR
181
C COMPILER
182
ASSEMBLER
183
LINKER
184
102
190
191
180
CPU
DEVICE
WORKING MEMORY
106
201
201
202
107

FIG. 2

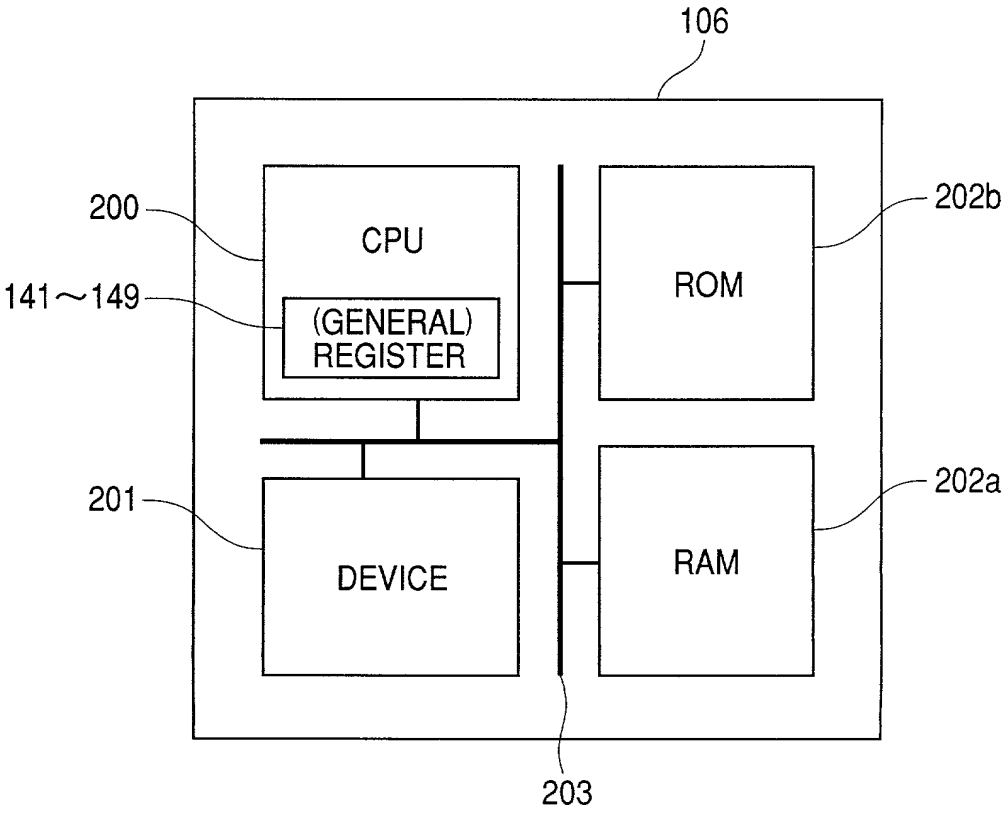
106
200
CPU
141～149
(GENERAL)
REGISTER
ROM
202b
201
DEVICE
RAM
202a
203

FIG. 3

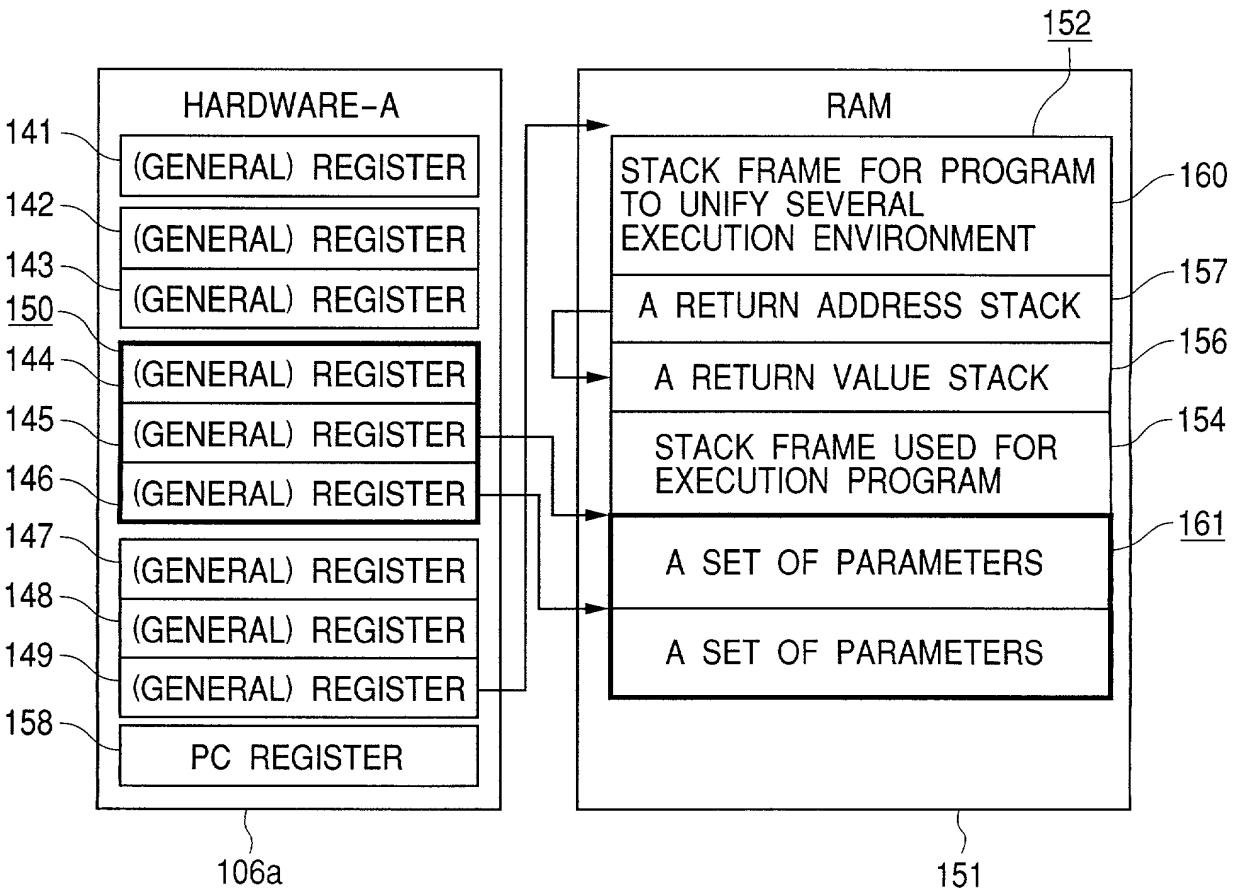
HARDWARE-A
141
(GENERAL) REGISTER
142
(GENERAL) REGISTER
143
(GENERAL) REGISTER
150
144
(GENERAL) REGISTER
145
(GENERAL) REGISTER
146
(GENERAL) REGISTER
147
(GENERAL) REGISTER
148
(GENERAL) REGISTER
149
(GENERAL) REGISTER
158
PC REGISTER
106a
152
RAM
STACK FRAME FOR PROGRAM TO UNIFY SEVERAL EXECUTION ENVIRONMENT
160
A RETURN ADDRESS STACK
157
A RETURN VALUE STACK
156
STACK FRAME USED FOR EXECUTION PROGRAM
154
A SET OF PARAMETERS
161
A SET OF PARAMETERS
151

FIG. 4

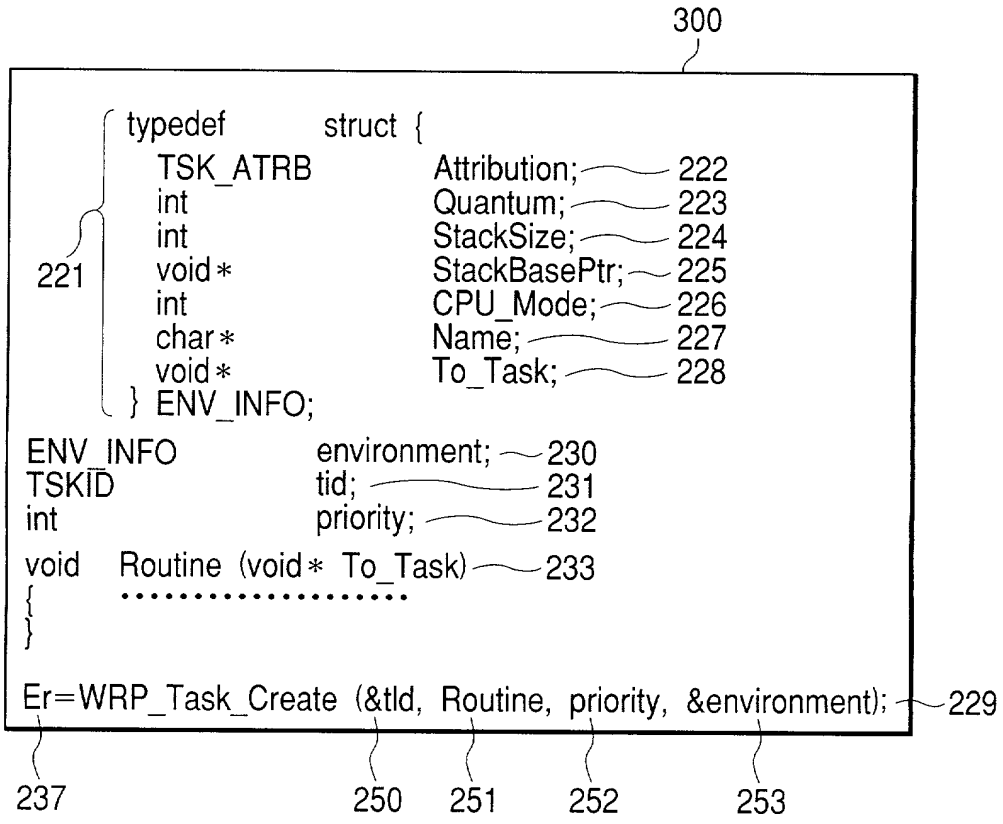
300
typedef struct {
TSK_ATRB Attribution; 222
int Quantum; 223
int StackSize; 224
221
void* StackBasePtr; 225
int CPU_Mode; 226
char* Name; 227
void* To_Task; 228
} ENV_INFO;
ENV_INFO environment; 230
TSKID tid; 231
int priority; 232
void Routine (void* To_Task) 233
{
}
Er=WRP_Task_Create (&tid, Routine, priority, &environment); 229
237
250
251
252
253

FIG. 5

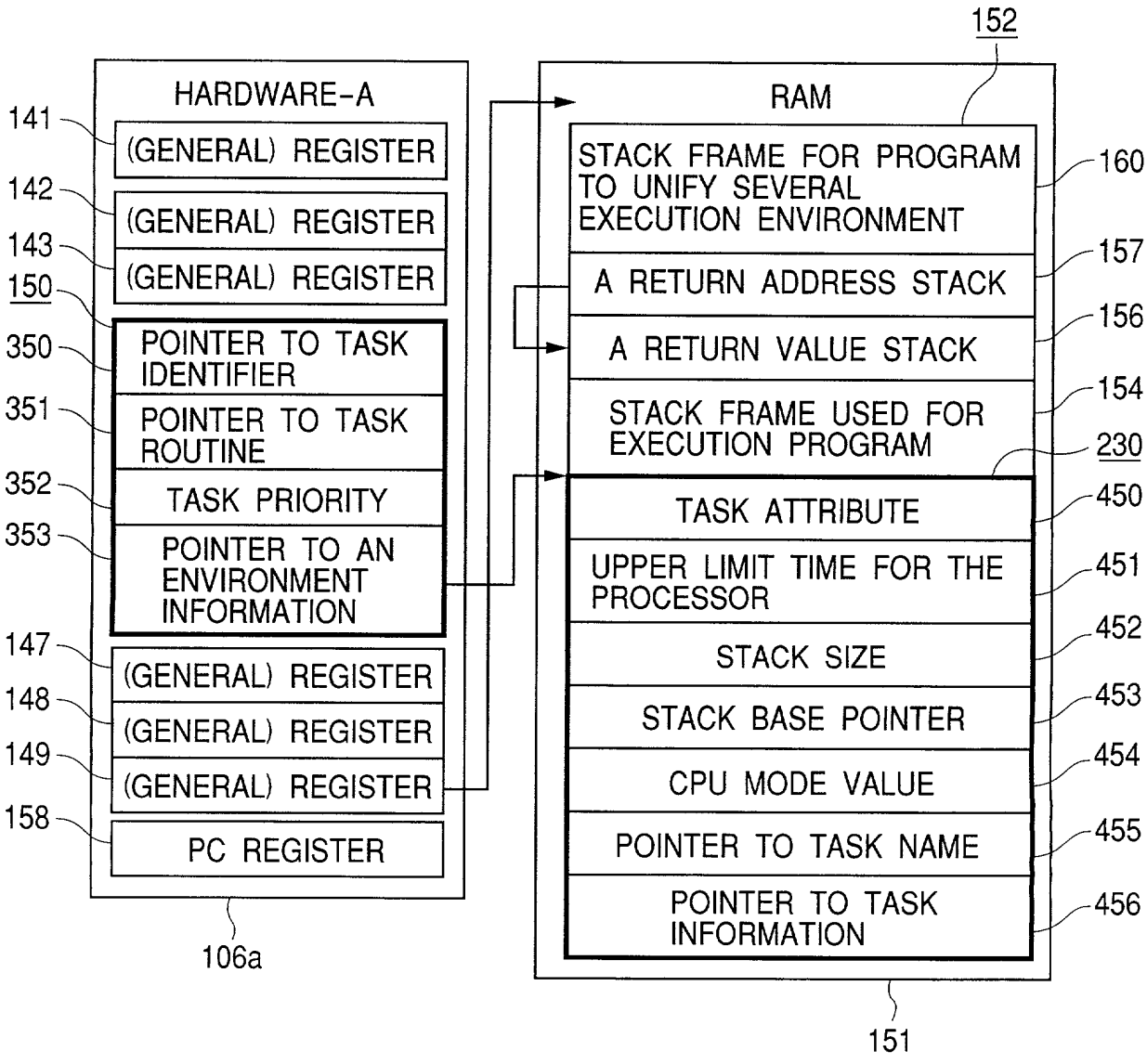
HARDWARE-A
141
(GENERAL) REGISTER
142
(GENERAL) REGISTER
143
(GENERAL) REGISTER
150
350
POINTER TO TASK IDENTIFIER
351
POINTER TO TASK ROUTINE
352
TASK PRIORITY
353
POINTER TO AN ENVIRONMENT INFORMATION
147
(GENERAL) REGISTER
148
(GENERAL) REGISTER
149
(GENERAL) REGISTER
158
PC REGISTER
106a
152
RAM
STACK FRAME FOR PROGRAM TO UNIFY SEVERAL EXECUTION ENVIRONMENT
160
A RETURN ADDRESS STACK
157
A RETURN VALUE STACK
156
STACK FRAME USED FOR EXECUTION PROGRAM
154
230
TASK ATTRIBUTE
450
UPPER LIMIT TIME FOR THE PROCESSOR
451
STACK SIZE
452
STACK BASE POINTER
453
CPU MODE VALUE
454
POINTER TO TASK NAME
455
POINTER TO TASK INFORMATION
456
151

FIG. 6 PRIOR ART

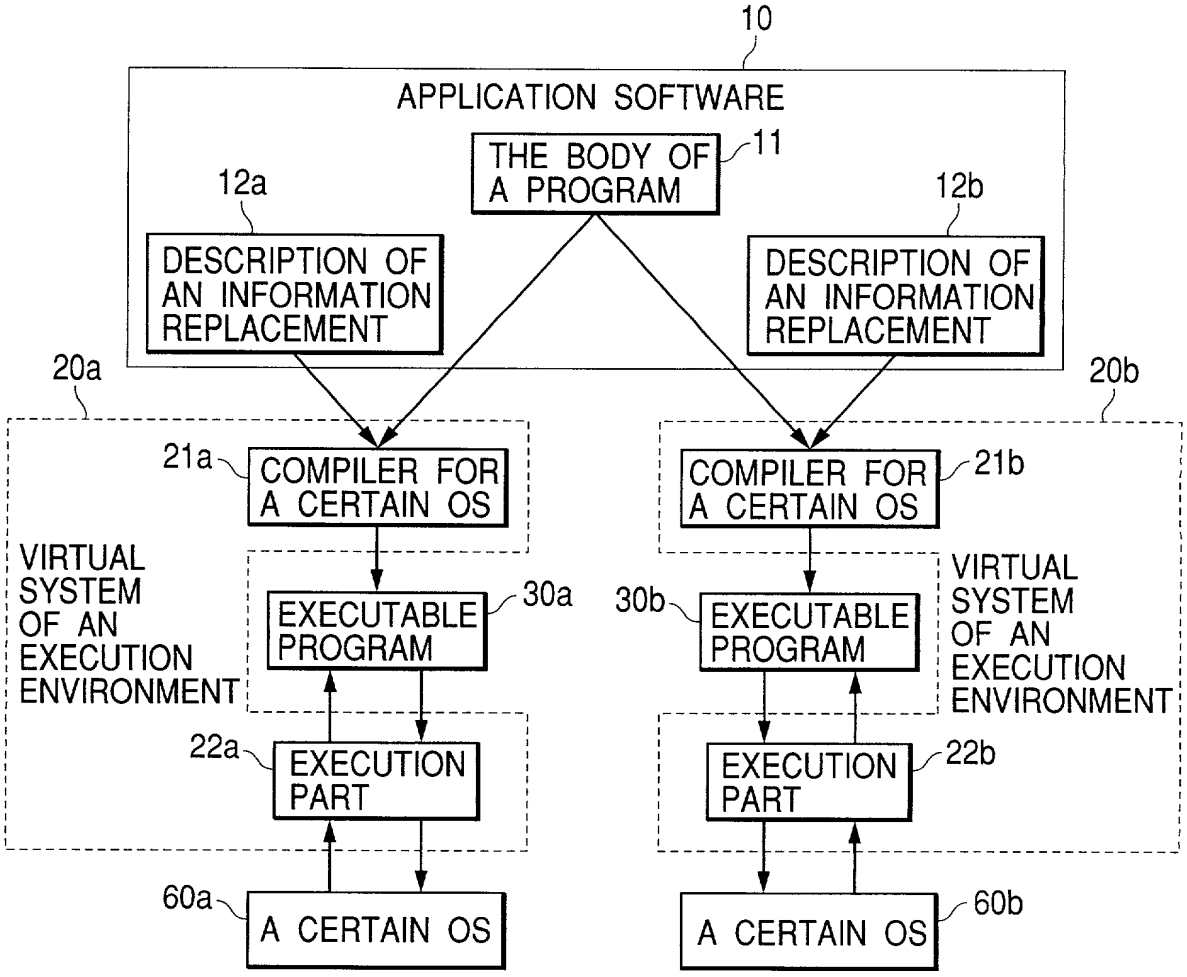

FIG. 7 PRIOR ART

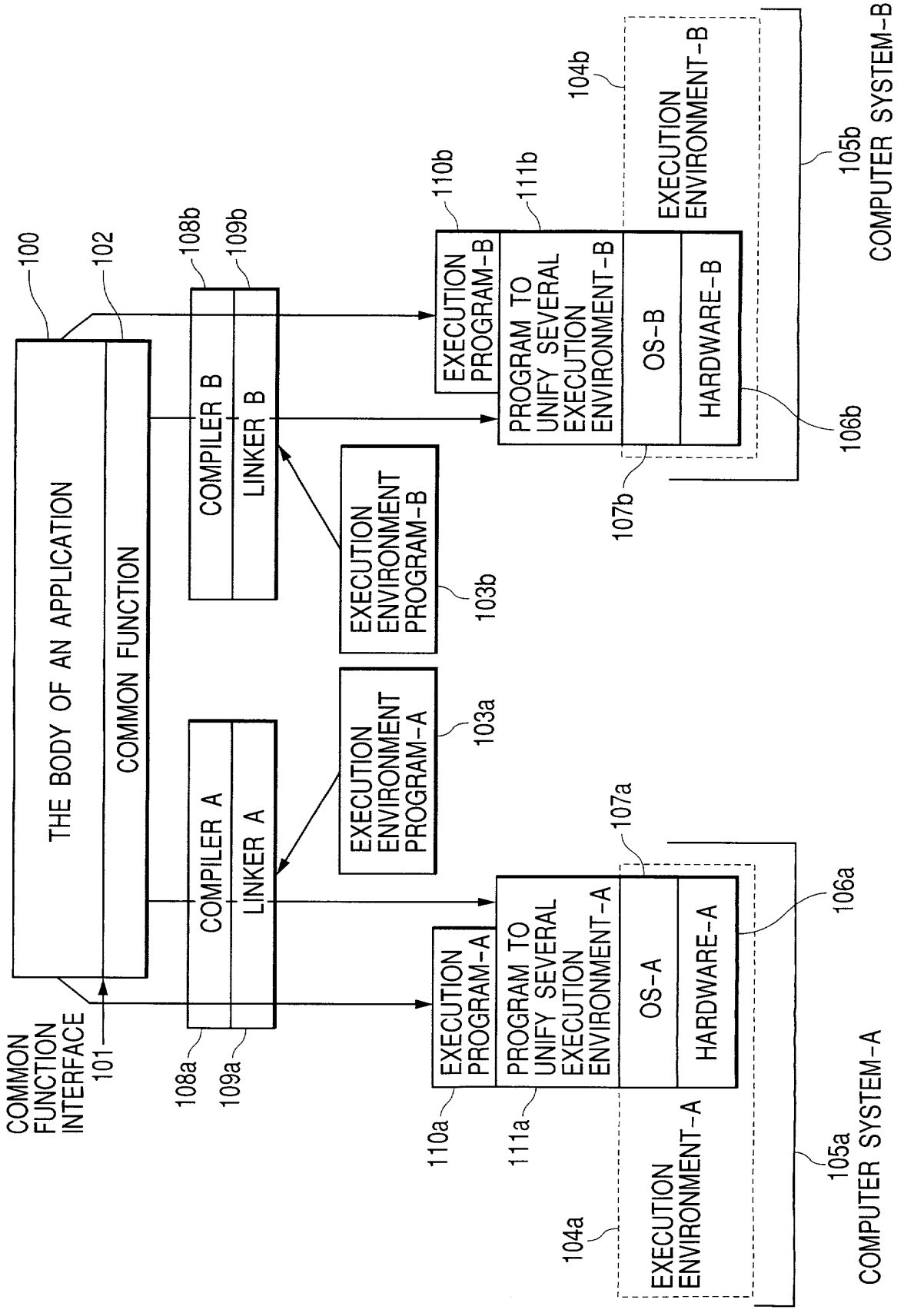

FIG. 8 PRIOR ART

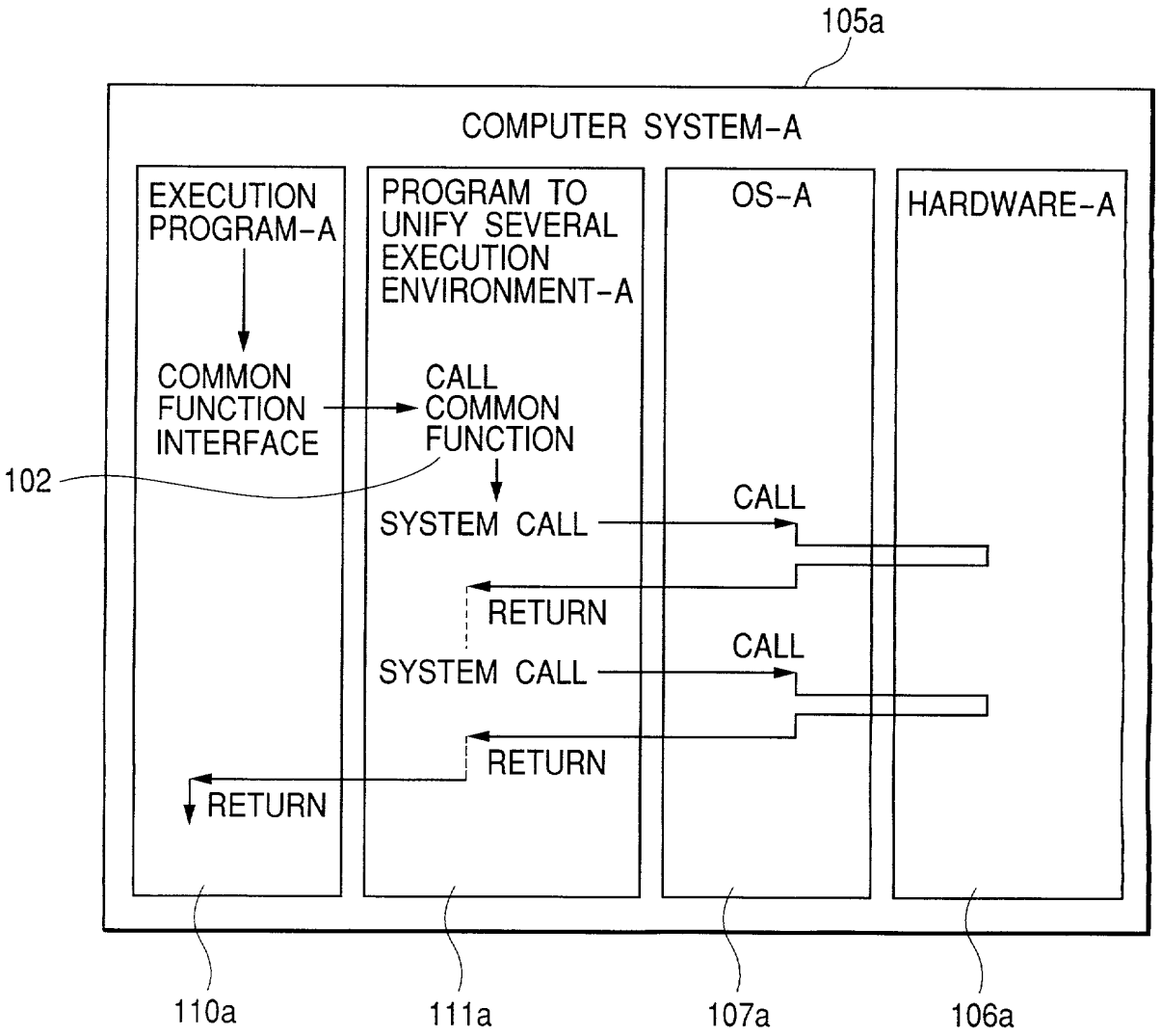
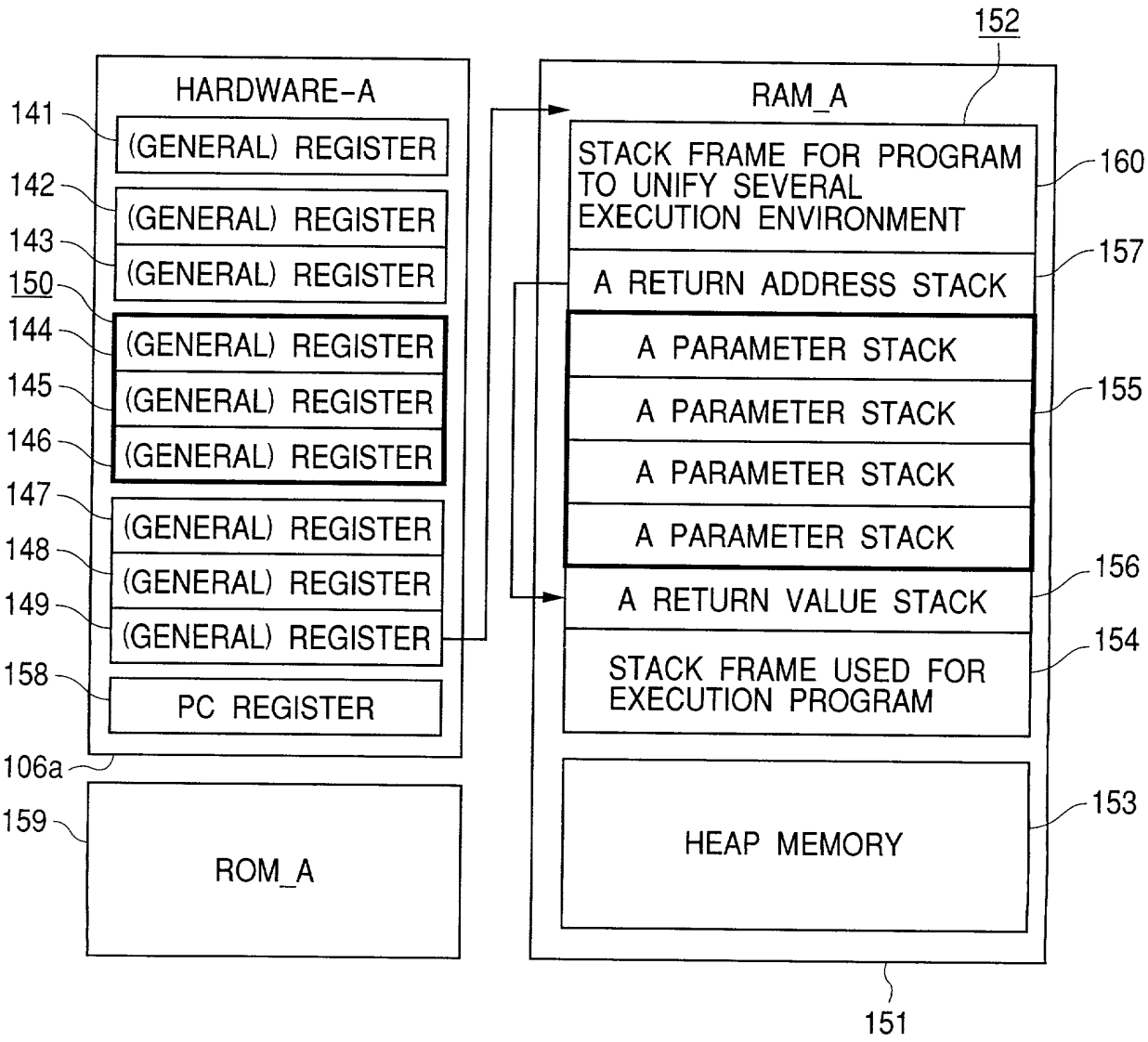
FIG. 9 PRIOR ART